Collaborating Across the Miles

Deb Agarwal

Ernest Orlando Lawrence Berkeley National Laboratory

Abstract

In today's scientific environment scientists can no longer work in isolated laboratories. The problems being studied require experts from several fields to combine their talents. The instruments used to gather scientific data are expensive and difficult to build so there are only a few in the world. The Internet connectivity and the powerful desktop workstations available today provide the potential to create collaboration tools and allow scientists to collaborate across the miles and access instruments remotely.

The DOE 2000 program sponsored by the US Department of Energy (DOE) is developing new technologies and integrating existing technologies to create a software environment that allows collaboratories to become a reality. The DOE 2000 program is composed of two pilot programs putting collaboratories into practice and five research and development projects developing some of the tools required to complete the collaboratory environment that were not already available. These technologies include videoconferencing over the Internet, electronic notebooks, shared visualization, security infrastructure, floor control, and session management. Much of this work is in conjunction with other DOE and non-DOE development efforts.

This paper describes the tools and technologies available today and some of the tools being developed to provide collaboration capabilities. Through collaboratories we are changing the way scientific experiments are carried out and the way scientists work together.

Introduction

The goal is to provide a location independent laboratory where scientists can collaborate. We refer to this type of laboratory as a *collaboratory*. A collaboratory presents the users with new ways of thinking, working with others, and doing tasks. For instance, with an electronic notebook data can be automatically recorded to the notebook as it is taken rather than hand recorded or printed and pasted into the notebook. Meetings can be held without needing a conference room or even needing to meet at the same site. A group that is collaborating but is separated geographically can use virtual office spaces to bring the group together into a common space.

For a collaboratory to be effective, it may need to provide remote access to any or all of the following: instrument settings, instrument controls, notebooks, logs, analysis results, meetings, and conversations. Paper notebooks and other components of the environment that are not currently computer applications may need to be converted to be computer accessible if they are to be accessed by remote users.

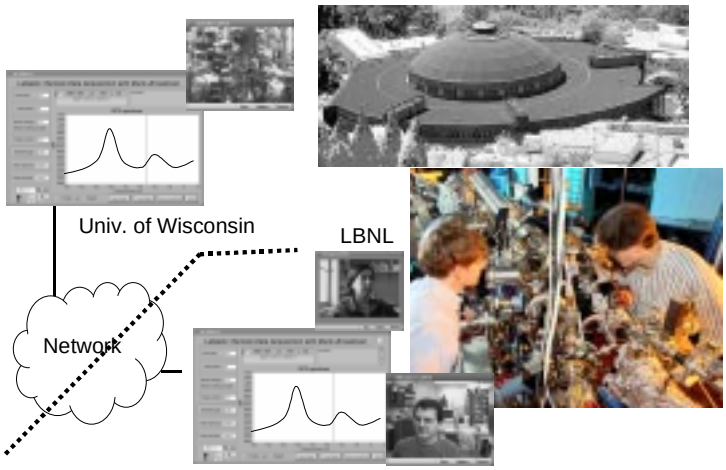

Figure 1. The Spectro-Microscopy Collaboratory at the Berkeley Lab Advanced Light Source

An important factor in building a collaboratory is scalability. Many of the early tools available for building collaboratories are based on unicast communication for distribution of information to the participating sites. Since unicast communication provides only point-to-point messages these tools have a server that is responsible for tracking the collaboratory members and redistributing all messages to the members. This type of server-based architecture is not readily scalable since the server and its network connection quickly become a bottleneck. Multicasting provides a more efficient means of communicating between collaborating sites since it allows group communication and the messages sent to the group are duplicated within the network only as needed to get to all the members of the group.

Background

There are many parallel development efforts designing and building collaboratories and the tools to support them. Some of the early efforts include remote electron microscopy[8] , the Upper Atmospheric Research Collaboratory[6] and the Distributed Collaboratory Experiment Environments Program (DCEE)[11] . The DCEE program was sponsored by the US Department of Energy and included development of four prototype collaboratories and several small research projects aimed at developing key technologies needed to support collaboratories. The prototypes include the Spectro-Microscopy Collaboratory[1] , shown in Figure 1, the Environmental and Molecular Sciences Laboratory[3] , the LabSpace project, and the Fusion Collaboratory[16] . The research projects worked on areas such as security, reliable multicast communication, electronic notebooks, and sociological issues of collaboratories. The DCEE prototype collaboratories have provided a basis for understanding how to build collaboratories and the issues involved in building a collaboratory.

Figure 2. The multicast-based videoconferencing tools. The tools from left to right are sdr, wb, vat, and vic.

The DOE 2000 program[12] is taking the collaboratory concept further by funding two large-scale collaboratory pilot projects and seven research projects. The pilot projects of the DOE 2000 program are the Diesel Combustion Collaboratory[10] and the Materials MicroCharacterization Collaboratory[9] . Both of these collaboratories include a large number of collaborators from a wide variety of institutions. They are addressing issues of sociology, scalability, network delay, data compatibility, and data conversion. The DOE 2000 research projects are building capabilities that have not previously been available. These research projects are providing continuing development of collaboratory tools and include projects focusing on: collaborative session management, electronic notebooks, shared virtual spaces, scalable security architecture, network quality of service, floor control, and a collaboratory interoperability framework. Tools under development by these research projects are now becoming available for integration into collaboratory environments in beta or alpha versions.

Tools

Each group of collaborators has a specific set of activities they need supported by a collaboratory so the tools have generally been developed to support individual activities. A collaboratory is built by using the tools that are most appropriate to the particular collaboration; the tools provide the collaboratory building blocks.

The first tool that most people think of adding to a collaboratory is videoconferencing. There has been significant commercial interest in videoconferencing and collaborative access to documents and workspaces. Several tools are now available from commercial and non-commercial sites. The multicast-based videoconferencing tools vic (video), vat (audio), rat (audio), sdr (session directory), and wb (whiteboard) tools, shown in Figure 2, are free and are scalable and cross-platform (not Macintosh)[14] . The multicast-based tools are used in many places to broadcast talks and videoconference across the Internet. Using videoconferencing tools alone leaves the remote user feeling like they are looking in a window into the room. The user needs instead to feel that they are present in the remote room, that they are able to participate in conversations, and that they can walk around. We refer to these capabilities as *telepresence*. Some tools have been developed to add telepresence capabilities to the multicast-based videoconferencing tools.

Figure 3. The conference controller and the remote camera control interface

These include a conference controller[5] that allows the videoconferencing tools to be controlled remotely and remote camera control software[15] (see Figure 3).

The Microsoft NetMeeting tool, shown in Figure 4, is available free from Microsoft and it provides unicast-based videoconferencing capabilities, whiteboard and limited application sharing. NetMeeting is only available for the Windows operating system. The application sharing capability in NetMeeting is based on a standard called T.120 and is interoperable with other T.120 compliant application sharing packages, but there are currently only a few of these available. Also, the T.120 standard is based on unicast messaging which does not allow peer-to-peer interaction within a group nor is it scalable to large groups.

If instead of sharing individual applications you would like to share an entire screen across multiple sites there is a shared screen viewer called Vitual Network Computer (VNC), shown in Figure 5. VNC is free software that provides cross-platform screen and application sharing capabilities[2,7]. It is not T.120 compliant so it does not interoperate with NetMeeting. VNC software is available in Java or as precompiled binaries so the number of platforms it works on is virtually unlimited. Since VNC was initially only designed to provide remote terminal display

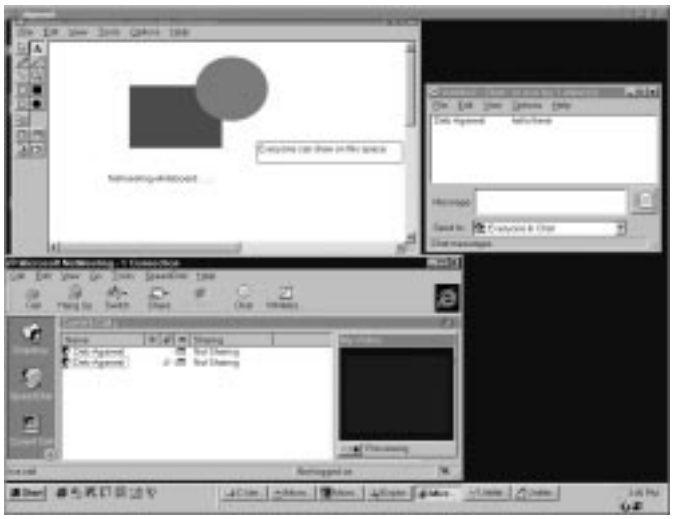

Figure 4. Microsoft NetMeeting videoconferencing and application sharing tool

Figure 5. Virtual network computing (VNC) screen sharing application. Shown here sharing a Microsoft Windows machine's screen onto a UNIX machine.

capabilities, it does not have control hand-off tools and instead allows all the sites sharing the interface to control the mouse and keyboard on a first-come-first-serve basis.

Several text-based interaction tools provide real-time interaction capabilities. These tools are relatively sophisticated and allow the text-based space to be partitioned into different rooms. Rooms can be used to segregate discussion topics. A user can also create a virtual office where they are available to collaborators but are not interrupted by other conversations. This spatial metaphor provides a means for the user to easily navigate the space using familiar methods of getting around like walking. These text-based tools are interacted with using a keyboard to type the conversation. They also allow the user to express emotions by doing things like smiling or chuckling. The text-based interaction tools are Internet Relay Chat (IRC) and the Multi-User Dungeons (MOO) developed at Xerox PARC[4] . Microsoft NetMeeting and NCSA Habanero[13] have chat facilities but these tools are limited in their functionality and do not implement the spatial metaphor.

Access to notebooks is an essential part of a scientific collaboratory. The scientific notebook is the place where events are recorded, results are displayed, and comments are written. Without access to the notebook a scientific collaborator is left to react to events rather than allowed to participate. There are several parallel development efforts cooperating to build electronic notebooks. These efforts are underway in both the commercial sector and as a DOE 2000 research project. The commercial sector is considering the legal aspects of electronic notebooks as invention records. The DOE 2000 project is developing prototype interfaces and engines for the notebook (see Figure 6).

There are several projects building components of the collaboratory infrastructure. The infrastructure, once it is available, will make the building of new collaborative applications significantly easier. The components will provide tools and abstractions for the tasks that most collaborative applications need to do such as floor control, shared data interchange, security, and group communication. These infrastructure components provide application programming interfaces to the collaboratory tool developers. These interfaces can be used to specify things like shared data, security, and application sharing properties. NCSA Habanero provides a Java-based framework that includes a session management component to provide the user interface glue and shared data abstractions. Sun's shared data toolkit provides a data sharing interface. The DOE 2000 Collaboratory Interoperability Framework provides a common communication

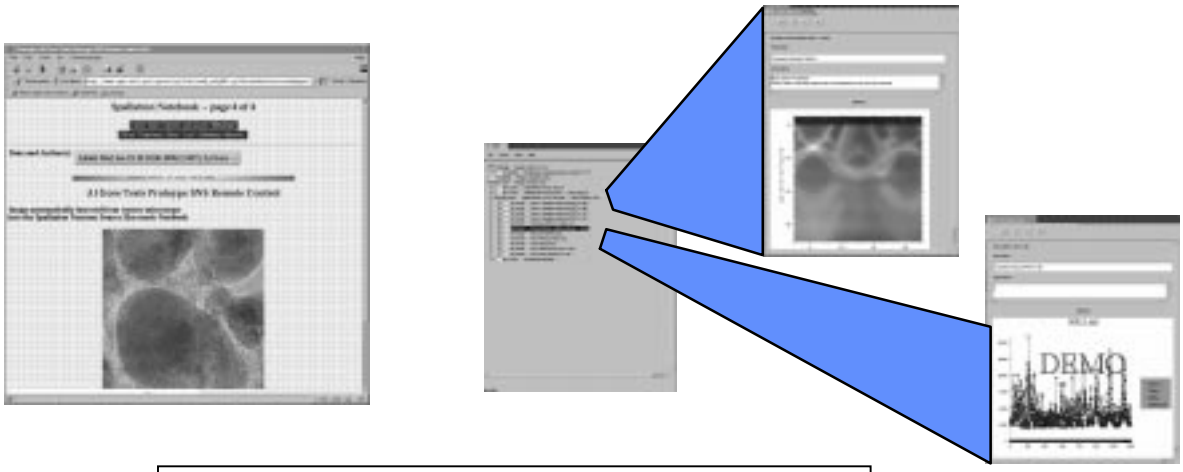

Figure 6. Two of the electronic notebook interfaces

library that includes multicast and unicast communication that includes reliable and unreliable multicast. It also provides directory services and a shared data abstraction.

Building a Collaboratory

When evaluating whether a group will benefit from a collaboratory, an essential requirement is a strong need for collaboration by the group members. This may seem like an obvious criteria, but many people envision a collaboratory as a way to get groups they would like to have working together to work together. The effort required to integrate collaboratory tools into the user environment and to customize them to the user's needs is non-trivial so the users should perceive considerable benefits from this effort.

The first step in building a collaboratory is an evaluation of the group's current work processes and collaborations. The goal is to identify a small number of collaboration tools to integrate into the environment. The tools should be chosen to have the largest benefit to the group with the least effort by the users to get them operational. It is also worth considering the level of intrusion into the environment, the tools should feel natural and appropriate to the tasks of the collaboration. Ideally, the collaboratory tools provide an improved functionality compared to the user's pre-existing tools.

To build a collaboratory you might assume that you should start by implementing videoconferencing. It has been our experience that this may not be a good choice. Videoconferencing capabilities require a considerable investment in time and materials to install. Each site that will participate in the videoconference will require video capture hardware and a sound card. If more than one person will participate from a single machine (e.g. a conference room) the system will also need to be equipped with an echo-canceller to allow hands-free use of the videoconferencing system. You might also want to consider the use of remotely controllable cameras.

We have generally found that a good place to start instead is with electronic notebooks and text-based interaction tools. We use the text-based interaction tools as a means to maintain a virtual office environment. Each user is logged into the space whenever they are on the network, providing a means of contacting colleagues that is a virtual equivalent to having the colleague be in the office down the hall. The text-based interaction tools are relatively non-intrusive, work

cross-platform, and are easy to learn. The next tools that need to be integrated are often audio-conferencing and application or screen sharing tools.

Critical application-specific software components that must be accessible to collaborators may need to be redesigned for the collaboratory. These applications have usually been designed for use by a single user at a single site. These applications can be shared by either rewriting them to be collaborative or by using application sharing software. Application sharing software takes a single-user interface and distributes it by sharing the display of the application and the mouse and keyboard inputs of multiple sites. The collaboratory infrastructure components simplify the rebuild of these applications.

Keep in mind that remote users introduce new problems with regard to software maintenance, user training and cross-platform support. The users may be unfamiliar with details of their system or software installation and upgrade procedures. Collaboratory tools need to be easy for remote users to install and maintain. Software needs to be self-installing, and resilient to diverse environments. Software upgrades need to be available over the Internet so that users can upgrade at their convenience (often only when incompatibilities force them to). Security and privacy constraints must also be considered in designing the collaboratory. Each site may have its own privacy and security constraints. Collaboratory tools are just beginning to have security features built-in.

Conclusion

The broad interest in using collaboratories has lead to increasing numbers of tools being available to support collaborative activities. As each tool becomes available there are new capabilities provided to the community and some set of users that adopt the tool. Each new set of users brings a fresh perspective to the collaboratory environment and helps provide feedback to the developers regarding what functionality is needed.

Since many of the tools are still in their infancy they are not always available for all platforms and are often not available as production software. Since there are few standards, many of the tools do not inter-operate effortlessly. The standards are slowly emerging with the tools and the interoperability is increasing daily. In the meantime, groups generally need to choose a tool and have all the users use the same version of the tool to ensure interoperability.

Some collaboratories may also require advanced networking infrastructure. If users are depending on remote access to components such as experiment equipment and need a minimum bandwidth to operate, some way of ensuring that the network is able to provide the service level required is needed. Other network services include capabilities such as directory services, security, and reliable multicast. Research projects are currently underway to address each of these issues and several solutions have been proposed and implemented. Standards are slowly emerging addressing each of these areas.

Although collaboratories are still in their infancy, there are already many tools available that allow groups to collaborate effectively. With a well considered choice of collaborative tools and a stepped integration plan, groups that are widely dispersed geographically can derive significant benefits from the increased contact and functionality provided by the tools. The level of group

connection that can be achieved even just through the use of a text-based interaction tool is astonishing.

References

[1] Agarwal, D. A.; Sachs, S. R.; and Johnston, W. E., ‘‘The Reality of Collaboratories,’’ *Computer Physics Communications* vol. 110, issue 1-3 (coverdate May 1998), pages 134-141.

[2] Li, S. F.; Hopper, A., “A Framework to Integrate Synchronous and Asynchronous Collaboration,” in Proceedings of IEEE Seventh International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE’98), Stanford University, June 17-19, 1998(ORL Technical Report 1998.12) (http:// www.orl.co.uk:80/abstracts.html)

[3] Myers, J.; Chonacky, N.; Dunning, T.; and Leber, E., “Collaboratories: Bringing national laboratories into the undergraduate classroom and laboratory via the Internet,” Council on Undergraduate Research Quarterly, vol. 17, no. 3 (March 1997).

[4] Mynatt, E. D.; Adler, A.;Ito, M.; O’Day, V. L. “Design for network communities,” Pemberton, S., editor, Human Factors in Computing Systems: CHI 97 Conference Proceedings; 1997 March 22-27; Atlanta, GA. New York, NY: ACM; 1997; 210-217 [MUD]

[5] Perry, M.A., “Confcntlr: A Videoconference Controller," Masters Thesis, San Francisco State University, December 1997. (available at http://www-itg.lbl.gov/~deba/ALS.DCEE/project.publications.html)

[6] Ross-Flannigan, N., "The Virtues (and Vices) of Virtual Colleagues," in *Technology Review*, Cambridge, MA: MIT, March/April, 1998. (available at http://www.techreview.com/articles/ma98/ross-flanigan.html)

[7] Wood, K. R.; Richardson, T.; Bennett, F.; Harter, A.; and Hopper, A.; “Global Teleporting with Java: Toward Ubiquitous Personalized Computing,” *IEEE Computer*, Vol 30, No 2, pp. 53-59, Feb 1997.

[8] Young, S. J.; Fan, G. Y.; Hessler, D.; Lamont, S.; Elvins, T. T.; Hadida, M.; Hanyzewski, G.; Durkin, J. W.; Hubbard,P.; Kindalman, G.; Wong, E.; Greenberg, D.; Karin, S. and Ellisman, M. H. (1996). “Implementing a Collaboratory for Digital Microscopy.” *International Journal of Supercomputing Applications and High Performance Computing*. 10(2/3): 170-181.

[9] Zaluzec, N.J., “Interactive Collaboratories for Microscopy & Microanalysis.” Multinational Conference on Electron Microscopy-97, Portoroz, Slovenia, Oct. 4-8, 1997, *Journal of Computer Assisted Microscopy*.

[10] Diesel Combustion Collaboratory (information available at http://www-collab.ca.sandia.gov/Diesel/Work/index.shtml)

[11] Distributed, Collaboratory Experiment Environments (DCEE) Program: Overview and Draft Final Report, February, 1997 (available at http://www-itg.lbl.gov/DCEE/Overview.fm.html)

[12] DOE2000: An Initiative to Transform DOE and Its National Laboratories, and Set the Standard for R&D Organizations into the 21st Century (available at http://www.mcs.anl.gov/DOE2000/ execsum.html)

[13] Habanero, (1998), "Habanero(TM) NCSA Home Page", (available at http://www.ncsa.uiuc.edu/SDG/Software/Habanero/)

[14] “Introduction to the Mbone,” (Information available at http://www-itg.lbl.gov/mbone)

[15] Remote Camera Control – devserv and camclnt (available at http://www-itg.lbl.gov/mbone/devserv/)

[16] Remote Experiment Environment – A Collaboratory (available at http://www.fusionscience.org/collab/REE/).